

## REMARKS/ARGUMENTS

On October 27, 2004, the attorney for applicants and the Examiners held a phone interview discussing the claims. Applicants discussed adding the requirements of claim 6 to independent claim 1, and similar amendments to the other corresponding independent claims, to clarify distinctions over the cited art. Although no agreement was reached, the Examiners said they would reconsider the rejection in view of such an amendment and argument distinguishing the amended claims from the cited art.

The Examiner rejected 1-9, 11-19, and 21-19 as anticipated (35 U.S.C. §102) by Popp (Pub No. US 2002/0133437). Applicant traverses the rejections for the following reasons.

Amended independent claims 1, 11, and 21 concern generating an interface to elements in a document, wherein the document defines a relationship of the elements and at least one attribute for each element. These claims require: providing a mapping indicating at least one element in the document to map to a class and an interface to generate for the class, wherein the interface defines methods to access the element for which the class is generated; and generating the class and the interface implementing methods for the at least one element from information provided on elements in the document and the mapping, wherein the at least one indicated element in the document for which the class is generated can be accessed and affected by the methods implemented in the class.

Applicants amended claims 1, 11, and 21 to include the requirement of providing a mapping indicating an element in the document to map to a class and an interface to generate for the class.

The Examiner cited pg. 2, paras. 0026-0027 and pg. 5, paras. 0069-0070 as disclosing the claimed mapping. (Final Office Action, pgs. 3). Applicant traverses.

The cited pg. 2 mentions how objects classes are defined for HTML elements in an HTML document, such that there is a one-to-one mapping of HTML elements to object classes. Although the cited pg. 2 mentions how object classes are created for HTML elements, which forms a mapping of objects to HTML elements, nowhere does the cited pg. 2 anywhere disclose an actual mapping data structure indicating at least one element in the document to map to a class and an interface to generate for the class that defines methods to access the element.

The claims 1, 11, and 21 require that there is a “mapping” data structure indicating elements to map to a class and an interface that is used to generate classes and interfaces

implementing methods for the elements. The cited pg. 2 discusses how a process of defining classes of objects for each HTML element has the effect of “providing a one-to-one mapping between each HTML element and object classes.” (Pg. 2, par. 0026). The cited pg. 2 is discussing how the result of generating the object classes produces or results in a mapping of elements to object classes. However, nowhere does the cited pg. 2 anywhere disclose or suggest a “mapping” data structure indicating elements and how each element maps to a class and an interface that are used to generate the class implementing methods as claimed.

The Examiner cited the above discussed pg. 2 pars. 26-27 and pg. 5, pars. 69-70 as disclosing the generating requirement (Final Office Action, pgs. 2-3), which as amended requires generating the class and the interface implementing methods for the at least one element from information provided on elements in the document and the mapping, wherein the at least one indicated element in the document for which the class is generated can be accessed and affected by the methods implemented in the class.

The cited pg. 5, paras. 0069-0070 of Popp discusses how the HTML template is parsed to generate an object tree based on identified HTML elements, and that an HTML object is instantiated for the objects in the object tree. Again, this cited pg. 5 shows how Popp teaches away from the claim requirement of using a separate “mapping” structure indicating at least one element in the document to map to a class to generate the classes implementing methods for the elements. Instead, Popp generates the object classes by parsing the document to generate an object tree.

Nowhere does Popp disclosing generating a class and interface implementing methods for one element. Instead, Popp discusses how object classes are defined HTML elements and that there are methods to manipulate the HTML elements mapped to object classes. (Pg. 2, par. 26). Nowhere does Popp disclose generating the class and interface implementing methods for one element as claimed.

Moreover, the cited Popp teaches away from using a mapping to generate classes for the elements. For instance, Popp discusses how an object tree of element objects is generated from the HTML document and that each object in the object tree corresponds to an object class which is instantiated. (Pg. 5, para. 0075.). Thus, Popp does not use a mapping data structure indicating elements to map to a class and interface to generate a class implementing methods for document

elements as claimed. Instead, Popp parses the HTML document to generate an object tree, and then generates the object classes from the objects in the tree.

The Examiner cited pg. 2, paras. 0026-0027 and pg. 3, para 0050 as disclosing the claim requirement that the at least one indicated element in the document for which the class is generated can be accessed and affected by the methods implemented in the class. (Final Office Action, pg. 3). Applicant traverses.

The cited pgs. 2 and 3 mention that “[e]ach object class can include methods to manipulate the HTML elements within an HTML document.” Popp mentions that each object class has methods that may be used to manipulate the HTML elements. Thus, in Popp, the object classes already have methods that can be used to manipulate the HTML elements. However, Popp does not disclose generating a class for each element implementing methods that can be used to access and affect the element in the document.

Moreover, Popp does not disclose and in fact teaches away from the claim requirement of generating the class implementing the methods, where the methods of the generated class can be used to access and affect the elements for which they are generated. Popp mentions that “[o]nce the object tree is generated, the HTML statements of an HTML document are generated by sending a ‘create’ message to the objects in the object tree.” (Pg. 6, para. 0081) Further, the “object tree is traversed during HTML document generation” (pg. 2, para. 0027).

Thus, Popp discusses using methods to access HTML elements in an object tree. Popp does not disclose using the generated object classes to access and affect elements in the document. Instead, Popp discusses generating an object tree, which is separate from the document, manipulating elements in the object tree, and then generating the HTML document from the object tree. Thus, in Popp the access and operations on the elements occurs through the element objects in the object tree, not within the document as claimed. For instance, Popp mentions that the “objects instantiated … are used to render the Web page definition (i.e., HTML document).” (Pg. 5, para. 0069)

Although the cited Popp does manipulate HTML elements that are from an HTML document, in operation Popp accesses and effects element objects in the object tree. Thus, Popp may discuss operating on elements from a document through the object tree, but nowhere does the cited Popp disclose accessing and affecting the elements in the document itself as claimed.

Accordingly, Applicant submits that amended claims 1, 11, and 21 are patentable over the cited art because Popp does not disclose all the claim requirements.

Dependent claims 2-10, 12-20, and 22-30 are patentable over the cited art because they depend from claims 1, 11, and 21, which are patentable over the cited art for the reasons discussed above. Moreover, the below discussed dependent claims provide additional grounds of patentability over the cited art.

Claims 2, 12, and 22 depend from claims 1, 11, and 21 and further require that the mapping includes a class name for each indicated element. The Examiner cited pg. 2, para 0026 and pgs. 4-5, para. 0063 of Popp as disclosing the additional requirements of these claims. (Final Office Action, pg. 3) Applicant traverses.

The cited Popp mentions that the name of the object class can mirror the name of the corresponding HTML element to provide an easy association. Although the cited Popp mentions how the object classes in the object tree can use the HTML element name, nowhere does the cited Popp disclose or suggest the claim requirement that a mapping indicating elements in the document to map to a class also include a class name for the element to map. Instead, the cited Popp uses the HTML element name for the class name, not a class name included in a separate mapping data structure that is used to generate the classes.

Accordingly, claims 2, 12, and 22 provide additional grounds of patentability over the cited art.

Claims 3, 13, and 23 depend from claims 1, 11, and 21 and further require that the mapping indicates a data type for at least one attribute of the indicated element. The Examiner cited pg. 1, para. 0014 and pg. 6, para 0079 as disclosing the additional requirements of these claims. (Final Office Action, pg. 3) Applicant traverses.

The cited pg. 1 mentions that HTML elements have properties, such as type. The cited pg. 6 mentions that a parser identifies a type of an element. Although the concept of type for elements is known, nowhere does the cited Popp anywhere disclose or suggest the claim requirement that a mapping indicating elements in the document to map to a class indicate a type for an attribute of an element to map to a class. Nowhere does the cited Popp anywhere disclose or suggest a separate mapping data structure that provides data type for attributes of elements. Instead, the cited pg. 6 of Popp has the parser identify the type from the current element statement in the document.

Accordingly, claims 3, 13, and 23 provide additional grounds of patentability over the cited art.

Claims 4, 14, and 24 depend from claims 1, 11, and 21 and further require that the relationship of the elements in the document are arranged in a hierarchical relationship, and wherein the methods in the at least one class generated for the element allow a user to directly access and affect the element in the document. The Examiner cited pg. 2, paras. 0024 and 0026 as disclosing the additional requirements of claims 4, 14, and 24. (Final Office Action, pgs. 3-4) Applicant traverses for the following reasons.

As discussed, the cited Popp mentions that the object calls includes methods to manipulate the HTML element in the HTML document. However, as discussed, in operation, Popp teaches away from accessing and manipulating the elements directly within the document and instead generates an object tree of objects that can be manipulated, so that the elements are affected through the object tree. In Popp, the HTML document is generated from the objects in the object tree. Nowhere does the cited Popp anywhere disclose or suggest allowing a user to use methods directly access and affect the element in the document. Instead, with Popp, the classes are used to manipulate the objects in the object tree or generate the HTML document from the objects, not the document itself as claimed.

Accordingly, claims 4, 14, and 24 provide additional grounds of patentability over the cited art.

Claims 5, 15, and 25 depend from claims 4, 14, and 24 and additionally require accessing the at least one element in the document indicated in the mapping using a hierarchical application program interface (API), wherein one class is generated for each accessed element. The Examiner cited pg. 4, para. 0062 as disclosing the additional requirements of these claims. (Final Office Action, pg. 4). Applicant traverses.

The cited pg. 4 mentions that objects corresponding to the HTML elements in the document are arranged in an object tree and rendering the HTML from the objects document methods of an object class to manipulate an HTML element. Although the cited pg. 4 mentions an object tree of objects corresponding to HTML elements, nowhere does the cited pg. 4 anywhere teach or suggest accessing an element in the document using a hierarchical application program interface, where there is one class for each element.

Accordingly, claims 5, 15, and 25 provide additional grounds of patentability over the cited art.

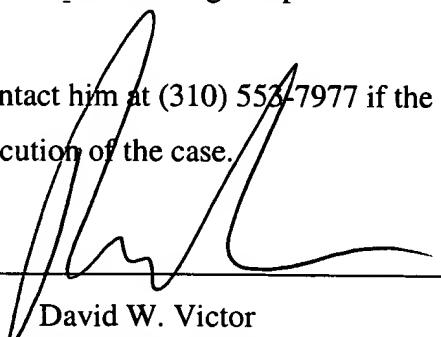
Claims 7-10, 17-20, and 27-30 are patentable over the cited art because they depend from claims 1, 11, and 21, which are patentable over the cited art for the reasons discussed above, and because their additional requirements in combination with the base and any intervening claims provide further distinction over the cited art. Moreover, the Examiner rejected claims 10, 20, and 30 as obvious over Popp in view of Skinner (U.S. Patent No. 6,085,198). However, Skinner was applied for the additional requirements of claims 10, 20, and 30, not the requirements of the base claims 1, 11, and 21 which in combination with the dependent claims provide still further grounds of patentability over the cited art.

#### Conclusion

For all the above reasons, Applicant submits that the pending claims 1-5, 7-15, 17-25, and 26-30 are patentable over the art of record. Applicant submits that no additional fee is needed. Nonetheless, should any additional fees be required, please charge Deposit Account No. 09-0460.

The attorney of record invites the Examiner to contact him at (310) 553-7977 if the Examiner believes such contact would advance the prosecution of the case.

Dated: December 13, 2004

By: 

David W. Victor  
Registration No. 39,867

Please direct all correspondences to:

David Victor  
Konrad Raynes & Victor, LLP  
315 South Beverly Drive, Ste. 210  
Beverly Hills, CA 90212  
Tel: 310-553-7977  
Fax: 310-556-7984